

WEB DEVELOPMENT



WEB DEVELOPMENT is the process of creating and maintaining websites and web applications that are accessible via the internet.

- **WEB:** It refers to websites, web pages or anything that works over the internet.
- **DEVELOPMENT:** It refers to building the application from scratch.

TYPES OF WEB DEVELOPMENT

- **Front-end development**
- **Back-end development**
- **Full-stack development**

Front End Development



FRONT – END DEVELOPMENT

- **Front-end developers use web technologies such as HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and JavaScript to build and enhance the user interface.**
- **HTML is used for structuring content, CSS for styling and layout, and JavaScript for adding interactivity and functionality to web pages.**
- **Front-end developers are also known as client-side developers, focus on the visual and interactive aspects of a website or web application.**

BACK-END DEVELOPMENT?



BACK – END DEVELOPMENT

- Back-end development, also known as server-side development, plays a crucial role in web development.
- Two key points about back-end development:
 1. Server-side logic
 2. Database Management



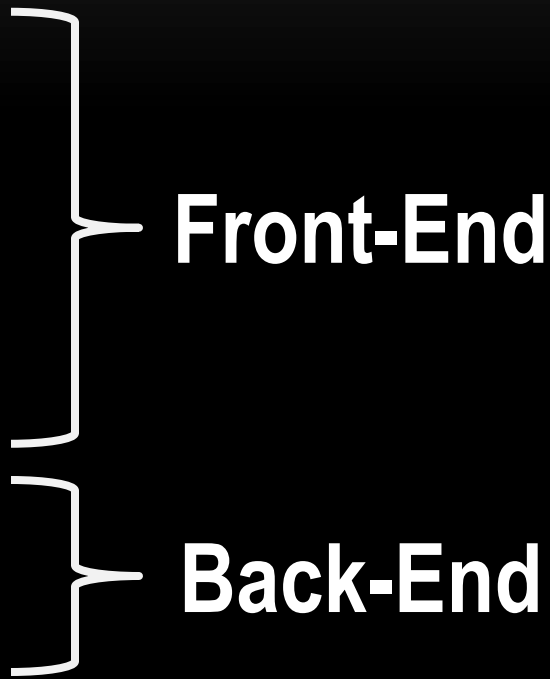
FULL STACK DEVELOPMENT

- Full-stack development is one comprehensive approach that covers both front-end and back-end aspects of web development.
- Full-stack developers can work on all aspects of a web application, from creating the user interface to managing the server and database.
- Two key points about back-end development:
 1. End to End Expertise
 2. Holistic Problem Solving

WEB DEVELOPMENT



WEB TECHNOLOGIES

- HTML
 - CSS
 - BOOTSTRAP
 - DBMS
 - REACT
 - PHP
 - JAVASCRIPT
- 
- The diagram uses white curly braces on a dark background to group the technologies. A large brace on the right side of the first five items (HTML, CSS, BOOTSTRAP, DBMS, REACT) is labeled 'Front-End'. A second brace on the right side of the last two items (PHP, JAVASCRIPT) is labeled 'Back-End'.
- Front-End
- Back-End

HTML

- **HTML (Hypertext Markup Language) is a fundamental language used for creating web pages on the internet. It consists of a set of codes, known as tags, that define the structure and content of a web page.**
- **HTML is responsible for organizing text, images, links, and other elements on a webpage, making it possible for web browsers to render and display content in a structured manner.**

HTML



DESIGN



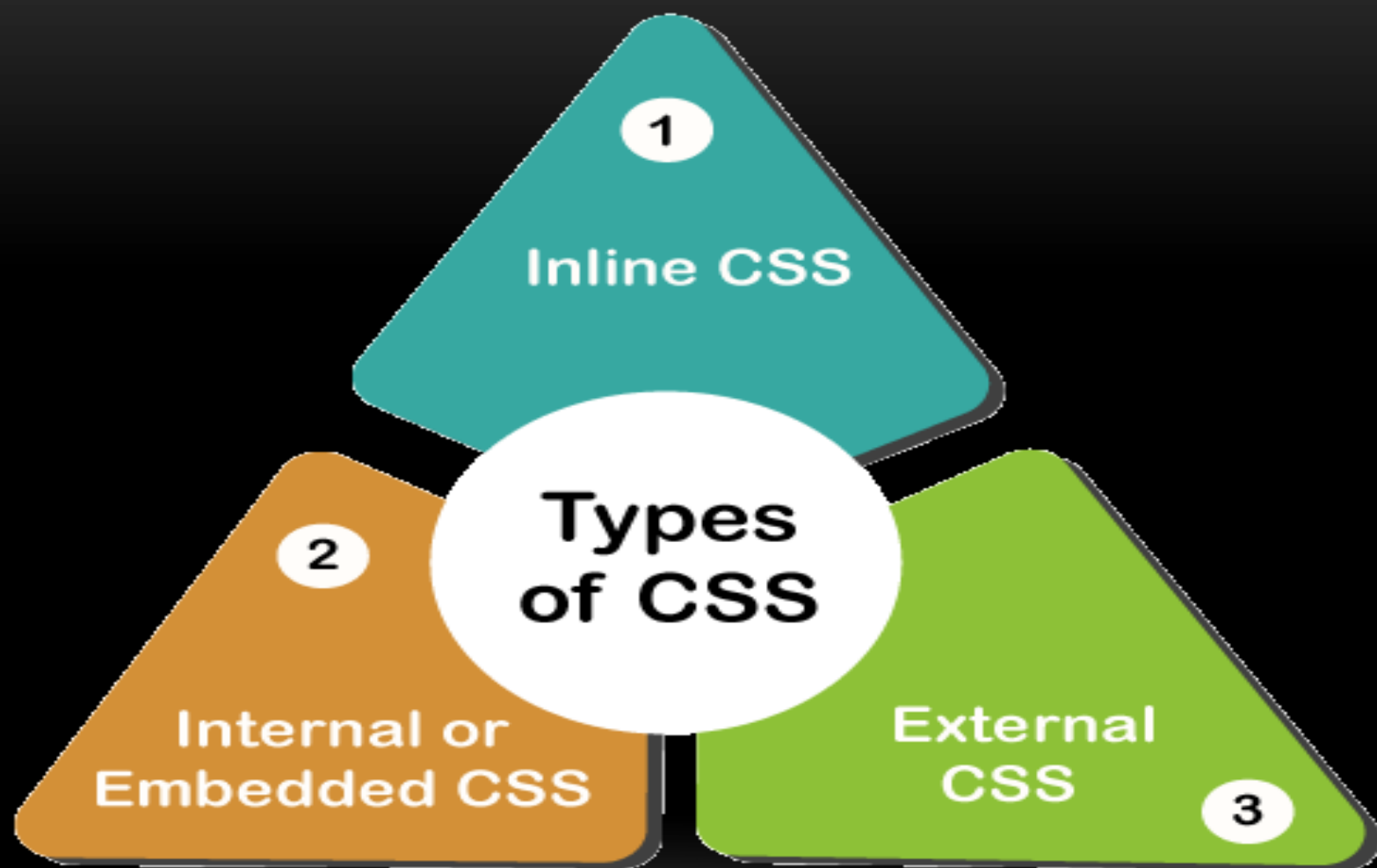
DEVELOPMENT



TESTING

CSS

- **CSS (Cascading Style Sheets) is a language used for enhancing the appearance of web pages.**
- **It allows web designers and developers to control the visual presentation of HTML elements by specifying details like colors, fonts, spacing, and layout.**
- **CSS separates the content (defined with HTML) from its visual design, making it possible to create attractive and consistent web pages. It's crucial for achieving a polished and professional look on websites by defining how elements are displayed in web browsers.**



BOOTSTRAP

- It provides a collection of pre-designed and customizable design elements, like buttons, forms, navigation bars, and grids, to help developers create responsive and visually appealing websites or web applications with less effort.
- Bootstrap simplifies the design process and ensures that websites look good on various devices and screen sizes by offering a consistent and mobile-friendly design foundation.



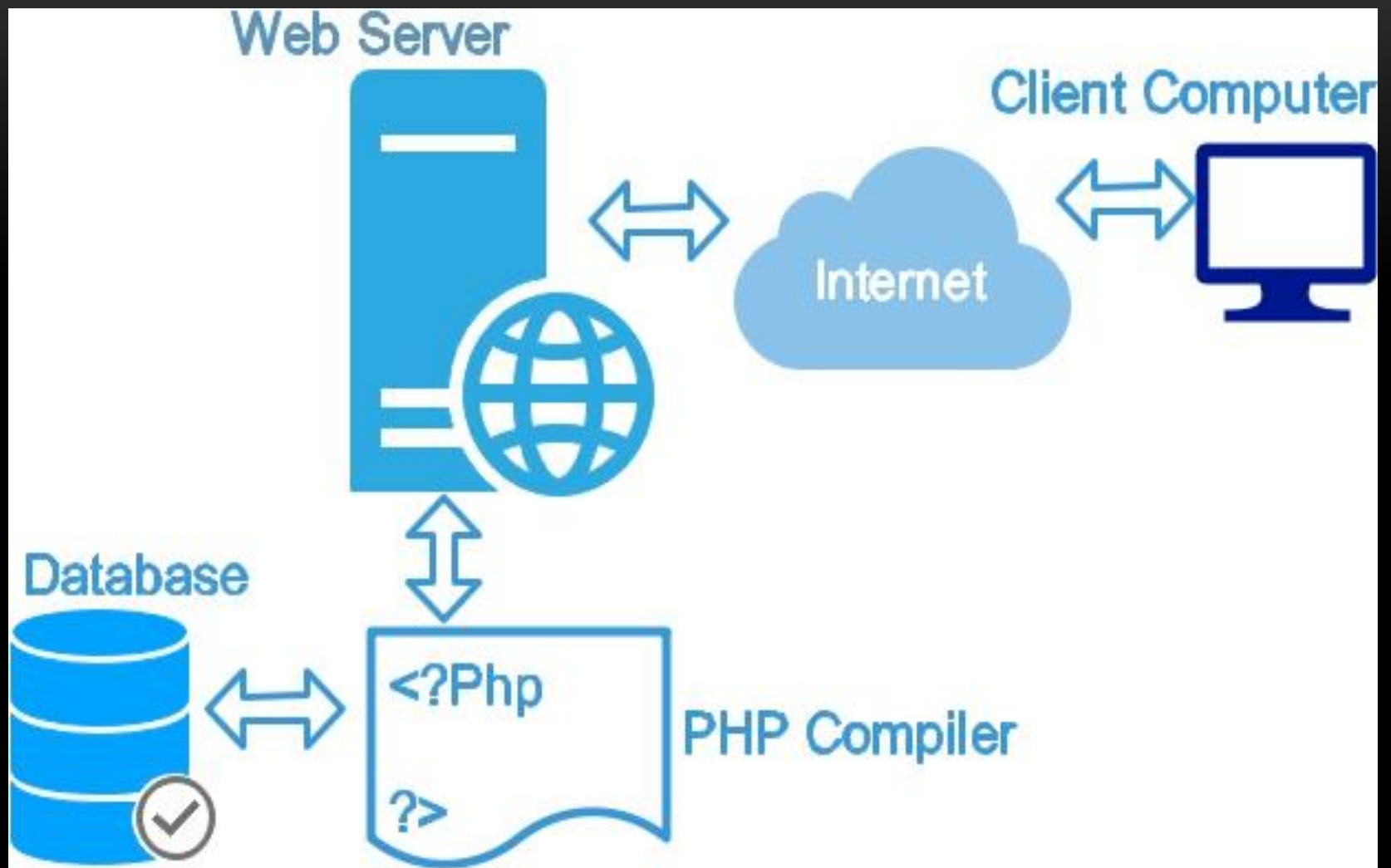
DBMS

- **Database Management System (DBMS) is a software tool that is used to organize and manage the storage and retrieval of data on a website. It enables developers to create, access, update, and manage databases where information such as user profiles, product details, and other content is stored.**
- **DBMS plays a crucial role in web applications by ensuring data is structured, secure, and easily accessible, facilitating tasks like user authentication, content management, and dynamic data retrieval.**



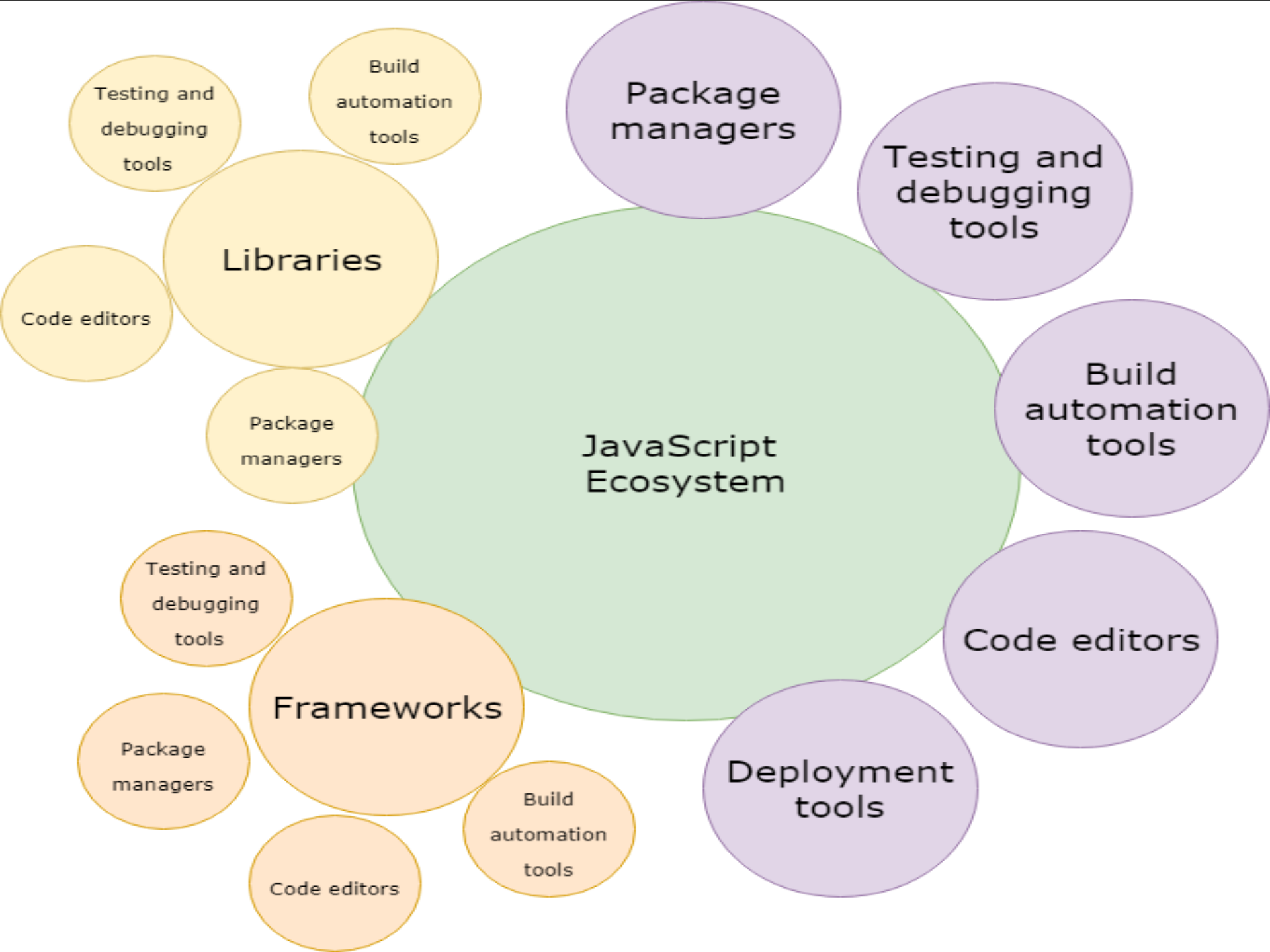
PHP

- **PHP (Hypertext Preprocessor) is a server-side scripting language used to create dynamic and interactive web pages. It processes code on the web server, generating HTML that is sent to the user's browser.**
- **PHP is essential for tasks like user authentication, database interaction, and server-side processing, enabling the creation of dynamic and data-driven web applications.**



JAVASCRIPT

- JavaScript is a versatile programming language primarily used on the client side. It runs in a user's web browser and is responsible for making web pages interactive and dynamic.
- JavaScript allows developers to create features like form validation, animations, and real-time updates on web pages, enhancing the user's experience.



REACT

- React is a JavaScript library used to build user interfaces. It's particularly popular for creating single-page applications (SPAs) and interactive web interfaces.
- React allows developers to build reusable, modular components that can update and display data in real-time, making it easier to create complex and responsive web applications.
- It simplifies the process of managing the user interface and handling user interactions.

PROS



Virtual DOM



Time taking on complex cases



Seo Friendly



Reusable Components

CONS



Improper Documentation



Easy data flow



JSX as a barrier



Easy debugging & testing



WEB APPLICATION

```
<!DOCTYPE html>
<html lang="en">

<head>
  <link rel="stylesheet" href="style.css">
  <script src="index.js"></script>
</head>

<body>
  <div id="clockContainer">
    <div id="hour"></div>
    <div id="minute"></div>
    <div id="second"></div>
  </div>
</body>

</html>
<style>
  #clockContainer {
    position: relative;
    margin: auto;
    height: 40vw;
    /*to make the height and width responsive*/
    width: 40vw;
    background: url(https://cutewallpaper.org/24x/feqo7zzu2/3220117606.jpg) no-repeat;
    /*setting our background image*/
    background-size: 100%;
  }
}
```

```
#hour,  
#minute,  
#second {  
position: absolute;  
background: black;  
border-radius: 10px;  
transform-origin: bottom;  
}
```

```
#hour {  
width: 1.8%;  
height: 25%;  
top: 25%;  
left: 48.85%;  
opacity: 0.8;  
}
```

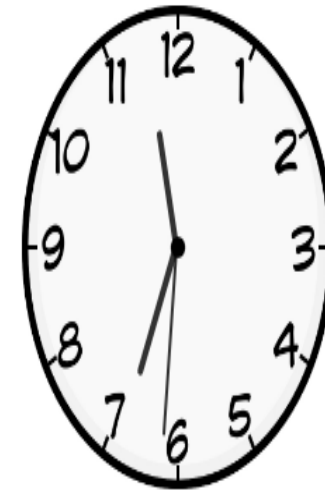
```
#minute {  
width: 1.6%;  
height: 30%;  
top: 19%;  
left: 48.9%;  
opacity: 0.8;  
}
```

```
#second {  
width: 1%;  
height: 40%;  
top: 9%;  
left: 49.25%;  
opacity: 0.8;  
}
```

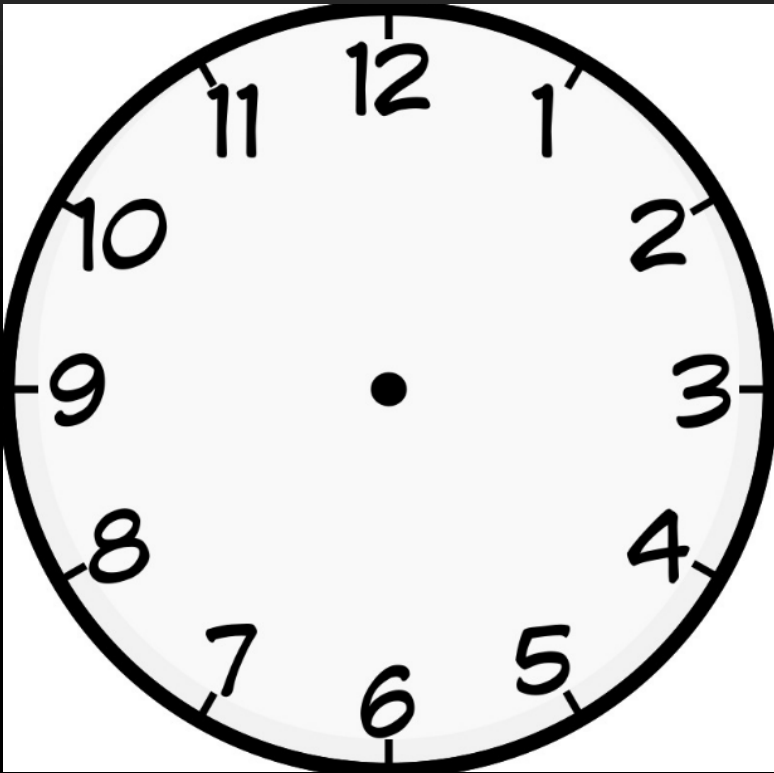
```
</style>
<script>
    setInterval(() => {
        d = new Date(); //object of date()
        hr = d.getHours();
        min = d.getMinutes();
        sec = d.getSeconds();
        hr_rotation = 30 * hr + min / 2; //converting current time
        min_rotation = 6 * min;
        sec_rotation = 6 * sec;

        hour.style.transform = `rotate(${hr_rotation}deg)`;
        minute.style.transform = `rotate(${min_rotation}deg)`;
        second.style.transform = `rotate(${sec_rotation}deg)`;
    }, 1000);
</script>
```

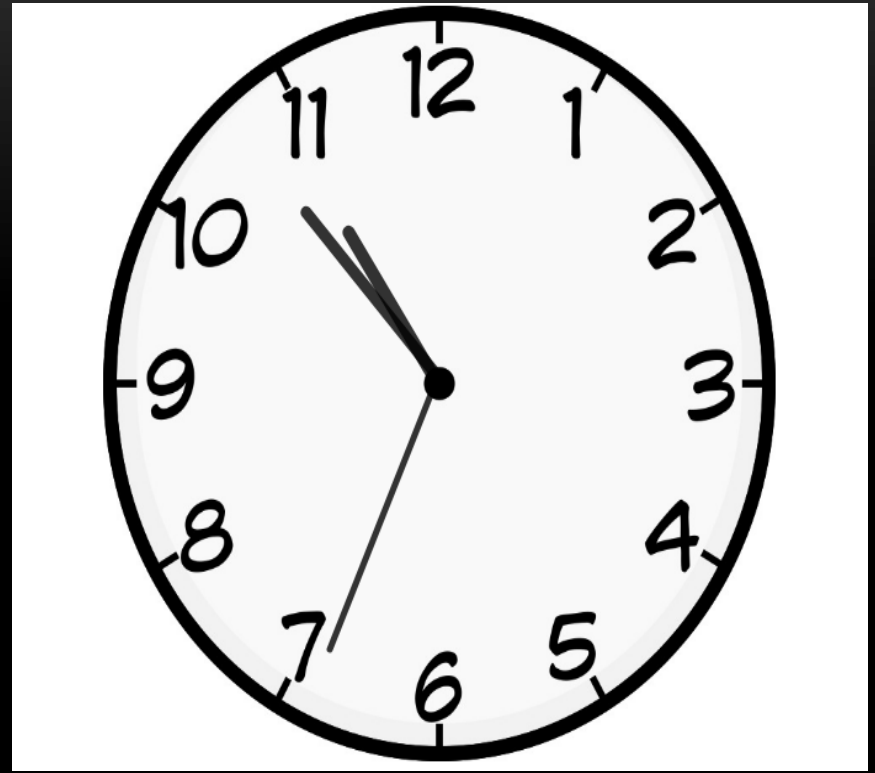
```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <link rel="stylesheet" href="style.css">
6   <script src="index.js"></script>
7 </head>
8
9 <body>
10  <div id="clockContainer">
11    <div id="hour"></div>
12    <div id="minute"></div>
13    <div id="second"></div>
14  </div>
15 </body>
16
17 </html>
18 <style>
19  #clockContainer {
20    position: relative;
21    margin: auto;
22    height: 40vw;
23    /*to make the height and width responsive*/
24    width: 40vw;
25    background: url(https://cutewallpaper.org/24x/feqo7zzu2/3220117606.jpg) no-repeat;
26    /*setting our background image*/
27    background-size: 100%;
28  }
29
30  #hour,
31  #minute,
32  #second {
33    position: absolute;
34    background: black;
```



Output



Before



After

CONCLUSION

My web development internship, which I found through Internshala Trainings, has been a valuable learning experience. Throughout this internship, I gained practical knowledge and hands-on experience in HTML, CSS, Bootstrap, PHP, JavaScript, React, and DBMS. These skills have not only helped me in developing various web applications but also provided me with a deeper understanding of web development.

THANK

YOU